

## **L'ordinateur : une innovation radicale (ou les deux machines de Turing<sup>1</sup>)**

**Michel CHEIN**

Académie des Sciences et Lettres de Montpellier

---

Cet article a paru dans les actes du colloque de la Conférence Nationale des Académies qui s'est tenu à Paris les 11 et 12 octobre 2019 : in Woronoff Michel (dir), *L'Innovation*, Akademos, Conférence nationale des Académies des lettres, Sciences et Arts, 2019, pp. 45-5

---

### **Introduction**

*(Les deux dialogues de cet article sont imaginaires et malgré tout plausibles, au moins en ce qui concerne les dates et les lieux de ces rencontres ainsi que leurs contenus scientifiques.)*

Il n'est pas de jour sans qu'une innovation informatique ne vienne profondément transformer notre société. Et ce, dans toutes les activités humaines, dans tous les domaines : économie (par ex. automatisation des processus industriels et de l'agriculture, organisation du travail, uberisation), commerce (par ex. commerce en ligne, systèmes de recommandation), finances (par ex. trading haute fréquence, bourse en ligne), médecine (par ex. imagerie médicale, robots médicaux, médecine à distance), transports, télécommunications, politique, vie privée, informatisation des processus administratifs, jeux et loisirs, ..., et dans toutes les sciences et techniques. Au cœur de tous ces bouleversements, que ce soit dans les réseaux sociaux, dans un télescope ou dans un téléphone, se trouve naturellement la science informatique et, matériellement, des ordinateurs. La première question qui se pose est donc : l'ordinateur est-il une innovation ou une invention ?

On dit souvent qu'une innovation est une invention qui a réussi. Au début est l'étincelle, l'idée, le verbe qui doit être transformé en création. Cette création, pour devenir innovation, doit encore marquer le monde, elle doit être acceptée par une partie de la population pour pouvoir se répandre dans la société et, généralement, devenir un objet de marché.

Toute invention peut être considérée comme une innovation potentielle et il n'y a pas de voie royale pour passer d'une invention à une innovation. Une innovation est souvent le résultat d'un assemblage, d'un bricolage, de plusieurs techniques ayant des origines diverses. Les relations entre invention et innovation sont loin d'être simples, et

---

<sup>1</sup> La première machine de Turing est la machine universelle décrite dans : A. TURING, *On Computable Numbers, with an Application to the Entscheidungsproblem*, Proceedings of the London Mathematical Society, London Mathematical Society, 1936, 230-265  
La deuxième machine de Turing est celle décrite dans : A. TURING, *Proposals for Development in the Mathematics Division of an Automatic Computing Engine (ACE)*, Report E882, Executive Committee, NPL

un court détour, par un ignorant du droit, dans le code de la propriété intellectuelle (en particulier son livre VI : Protection des inventions et des connaissances techniques) ou sur le site de l'Institut National de la Propriété Industrielle (INPI) ne permet pas de lever toutes les ambiguïtés. En effet, les deux premières étapes du dépôt d'un « brevet d'invention » – titre de propriété délivré par l'INPI ayant pour objet d'encourager l'innovation – sont les suivantes :

1. Vérifiez l'état de la technique et gardez le secret sur votre invention
2. Vous assurez (sic) que votre innovation est brevetable<sup>2</sup>

De plus, difficulté supplémentaire, une innovation n'est pas nécessairement la création d'un objet matériel, en particulier lorsque l'on considère des innovations dans les arts, et on pourrait dire, en détournant Roland Barthes<sup>3</sup> : « innovation/invention : terminologiquement, cela vacille encore, j'achoppe, j'embrouille. De toute manière, il y aura toujours une marge d'indécision ; la distinction ne sera pas source de classements sûrs, le paradigme grincera, le sens sera précaire, révoquant, réversible, le discours sera incomplet. »

Cependant, quels que soient les critères retenus, l'ordinateur – objet original de marché, diffusé et accepté par toute la société – est une innovation radicale, dans le sens où, comme dans le cas de l'imprimerie, elle a changé le monde. « Notre monde devient numérique »<sup>4</sup>, nous sommes entrés dans « le temps des algorithmes »<sup>5</sup>, on pourrait dire dans celui de l'informatique si l'on veut mettre l'accent sur la science sous-jacente à cette numérisation<sup>6</sup> ou dans celui des ordinateurs si l'on met l'accent sur l'objet matériel, présent partout, l'innovation radicale de notre époque<sup>7</sup>.

## La (première) Machine de Turing

(Princeton 1936)

Un jeune homme de 24 ans – chemisette froissée dont un pan sortait d'un short long dont les jambes arrivaient sous le genou, pieds nus dans des tennis blanches, cheveux bruns, lisses, coiffés avec une raie à gauche, regard vif –, n'avait fait que les premières foulées de son parcours quotidien de 4 miles dans les magnifiques parcs du campus, lorsqu'il fut interpellé par un homme grand, avenant, souriant, grosses lunettes d'écailles, habillé (costume-chemise blanche-cravate), une serviette sous le bras, sortant du Nassau Hall.

- Bonjour Alan, comment allez-vous ?
- Très bien, merci, j'ai croisé Albert Einstein il y a quelques instants et il fait tellement beau ce matin, et vous-même Professeur ?
- Ouf ! Les formalités sont accomplies. Je suis officiellement votre superviseur de thèse. Nous allons être amenés à travailler ensemble alors, s'il vous plait, ne m'appellez plus Professeur (même si j'ai une dizaine d'années de plus que vous, je

<sup>2</sup> Extrait le 17 avril du site : <https://www.inpi.fr/fr>

<sup>3</sup> R. Barthes, *Le plaisir du texte*, Paris, Éditions du Seuil, 1973, p. 10

<sup>4</sup> G. Berry, *L'Hyperpuissance de l'informatique*, Odile Jacob, 2017

<sup>5</sup> S. Abiteboul et G. Dowek, *Le temps des algorithmes*, Le Pommier, 2017 (si on n'a pas le temps de lire la référence précédente)

<sup>6</sup> M. Chein, *L'informatique : la science au cœur du numérique*, Bull. Acad. Sc. Lett. Montp., vol. 48, 2017, 203-214 (si on n'a pas le temps de lire la référence précédente)

<sup>7</sup> M. Nivat <http://www.epi.asso.fr/revue/articles/a1512c.htm> (un court article décapant de M. Nivat sur la propagande numérique)

ne me sens pas si vieux que ça), sinon je serais obligé de vous appeler *Monsieur Alan Mathison Turing*, ce serait un peu ridicule, très vieille Europe, n'est-ce-pas ?

– Comment dois-je vous appeler ?

– Réciprocité, ou symétrie, comme vous voulez Alan.

– Très bien Alonzo.

– J'ai étudié votre machine hier au soir ...

– Ma machine ?

– Le modèle mathématique d'une fonction calculable, que vous proposez dans votre manuscrit sur le *Entscheidungsproblem*, est très élégant. Voudriez-vous le présenter au prochain séminaire ?

– Bien sûr ! Mais, machine ... quand même. Je proposais un modèle abstrait de quelqu'un en train de calculer à la main avec un crayon et une feuille de papier. Il a devant lui un ruban de papier découpé en cases sur lequel ...

– Je sais, je sais, j'ai lu avec attention votre travail, et vous verrez Alan, dit en riant Alonzo Church, les « machines de Turing » deviendront plus populaires que mon lambda-calcul ! Il m'a permis de résoudre, comme vous, par la négative, le « problème de décision »<sup>8</sup> de Hilbert, de plus, les deux notions de fonctions calculables sont, je pense, équivalentes.

– Nous en avons parlé avec Kleene. Ce serait intéressant de prouver que ses fonctions récursives sont aussi équivalentes à nos fonctions calculables. Nous aurions, enfin, après plusieurs millénaires d'usage, une définition mathématique solide pour cette notion d'algorithme.

– Pauvre Hilbert, après Gödel, nos résultats vont l'achever !

– Je pense qu'il regrette Berlin et est plus abattu par les succès du régime nazi que par ...

– Je plaisantais, Alan, je plaisantais. Sérieusement, quand j'ai lu votre travail, j'ai immédiatement pensé au métier à tisser de Jacquard ou à un orgue de Barbarie (quel drôle de nom ...). En entrée, un ruban avec des symboles (ou des trous), une tête de lecture qui en fonction du symbole lu (ou de la présence d'une configuration de trous) envoie de l'air dans certains tuyaux puis passe au symbole suivant ou s'arrête.

– Ma tête de lecture contrôle non seulement le mouvement du ruban mais elle peut aussi changer le symbole lu. Et ...

– Certes, certes ... ne vous fâchez pas Alan, vous êtes bien sérieux ce matin, ce n'est qu'une analogie grossière, et votre notion clé, que je propose d'appeler *machine de Turing universelle*, permet non seulement de résoudre d'une manière simple et élégante le problème de décision, mais est, en quelque sorte, un automate permettant d'effectuer n'importe quel algorithme (ce que vous appelez une table, un ensemble fini de règles, un programme si on parle comme Babbage). Inutile de construire un automate pour chaque algorithme, on fournit en entrée une description d'un algorithme et les données sur lesquelles on veut l'exécuter et votre machine universelle fait le travail, c'est une machine superbe !

Turing, encore un peu vexé que Church ait publié, quelques semaines avant lui, sa preuve de la résolution du problème de décision, se demandait si le terme de *machine* pour la notion qu'il avait introduite pour résoudre ce problème n'avait pas des connotations péjoratives. Il présentait la notion que Church fera passer à la postérité en la nommant *machine de Turing*, comme un modèle abstrait d'une personne en train de

---

<sup>8</sup> A. CHURCH, *A note on the Entscheidungsproblem*, Journal of Symbolic Logic, 1, 1936, pp 40–41

réaliser un calcul avec un crayon et une feuille de papier. Imaginez-vous devant un ruban de papier quadrillé. Un nombre fini cases sont remplies (chaque case par un symbole), les autres étant vides (pour ne pas être gêné par la taille du ruban on considère que le ruban est infini, on a donc une infinité de cases vides). Vous avez un problème à résoudre dont les données sont représentées par les cases initialement remplies. Vous regardez une case et suivant son contenu et l'état d'avancement du problème que vous voulez résoudre, vous pouvez faire l'une des opérations simples suivantes : remplacer ou non le caractère lu par un autre et passer à la case à gauche ou à droite, et c'est tout (et ne faire ces opérations qu'un nombre fini de fois).

Une telle machine est donc un mécanisme très simple et la thèse de Church-Turing dit que : tout algorithme (tout traitement mécanique d'une séquence de caractères) peut être réalisé par une machine de Turing.

En guise d'exemple (dont la lecture n'est pas essentielle pour le propos de cet article, un lecteur pressé peut aller directement à la section suivante), voici une machine de Turing permettant de résoudre un problème très simple consistant à remplacer dans un mot (une succession finie de lettres, pas nécessairement un mot d'une langue) les minuscules par des majuscules (sans changer les lettres majuscules présentes initialement dans le mot). Deux états sont suffisants, un état de travail noté par T et un état final noté par F.

Pour simplifier nous allons écrire une machine de Turing faisant ce travail sur un mot ne contenant que les trois premières lettres de l'alphabet. Le fonctionnement de la machine consiste à appliquer les règles ci-dessous tant que c'est possible (un tel ensemble de règles est ce que Turing appelle une table, on peut aussi le voir comme un programme écrit dans un langage déclaratif).

**Si** on lit a dans l'état T **alors** on remplace a par A, on lit la case droite et on reste en T

**Si** on lit b dans l'état T **alors** on remplace b par B, on lit la case droite et on reste en T

**Si** on lit c dans l'état T **alors** on remplace c par C, on lit la case droite et on reste en T

**Si** on lit A dans l'état T **alors** on lit la case droite et on reste en T

**Si** on lit B dans l'état T **alors** on lit la case droite et on reste en T

**Si** on lit C dans l'état T **alors** on lit la case droite et on reste en T

**Si** on lit une case vide dans l'état T **alors** on passe dans l'état F

Initialement on lit la première lettre du mot, la plus à gauche, et on est dans l'état T. Puis on applique les règles tant que c'est possible (on s'arrêtera quand on aura parcouru le mot de la gauche vers la droite et qu'on lira la première case vide).

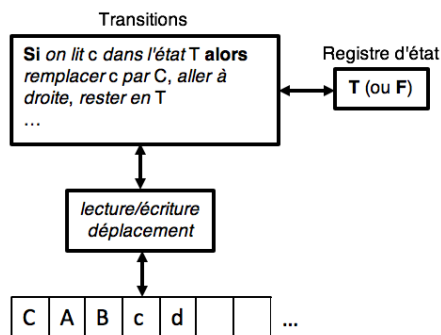


Figure 1. Une machine de Turing

Sur le schéma de la Figure 1, cette machine est représentée avec la seule règle de transition qui peut s'appliquer quand on lit la lettre c. L'application de cette règle remplacera c par C et on se déplacera sur la droite pour lire d. Après avoir remplacé d par D, le calcul s'arrêtera puisqu'on lira une case vide et que la seule règle applicable lorsqu'on lit une case vide conduit à l'état final F.

## Machine de Turing universelle

Turing a également défini ce qu'on appelle une *machine de Turing universelle*. Si la description d'une machine de Turing universelle n'est pas très simple, son fonctionnement est simple. Une machine de Turing universelle U est une machine qui, lorsque sur son ruban sont écrites une machine de Turing M (comme celle que nous venons de voir) et une donnée D (comme un mot composé de minuscules et majuscules), produit comme résultat celui que fournit la machine M dont le ruban contient initialement D. La machine est dite universelle car elle fonctionne de la sorte quels que soient M et D.

Ceci peut s'écrire symboliquement de la manière suivante : si  $M(D)$  représente le résultat du fonctionnement de la machine de Turing (simple) M sur la donnée D et si  $U(M, D)$  représente le résultat du fonctionnement de la machine universelle U sur la donnée (M, D) alors  $U(M, D) = M(D)$ . Le schéma de la Figure 2 illustre ce fonctionnement.<sup>9</sup>

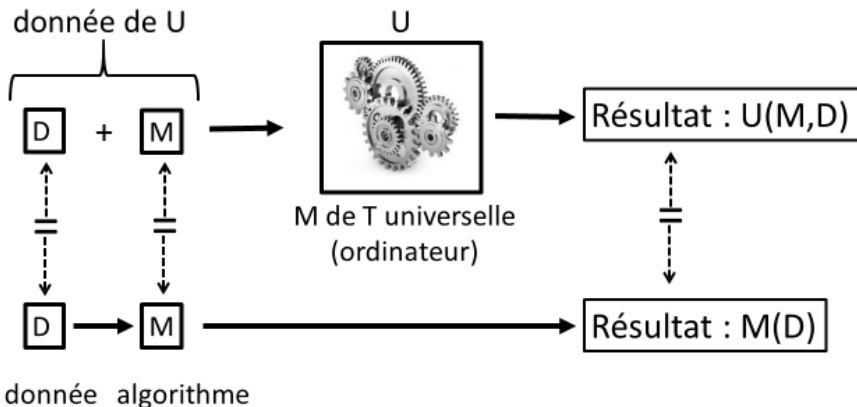


Figure 2. La machine de Turing universelle

Qu'est-ce qu'une machine qui, étant donné un programme (une machine de Turing) et une donnée mis en mémoire (sur le ruban), exécute le programme avec cette donnée ? C'est un ordinateur ! Un ordinateur est donc une machine de Turing universelle, on peut aussi dire qu'un ordinateur n'est qu'une machine de Turing universelle et, comme une machine de Turing ne semble pas très maline par rapport à ce qu'un être humain sait faire on peut s'interroger sur *l'intelligence* des ordinateurs...

<sup>9</sup> Pour résoudre le problème de décision, Turing démontre qu'il est impossible de définir une machine qui pourrait décider que pour n'importe quel (M, D) le calcul s'arrête ou ne s'arrête pas.

## De vraies machines

Brian Randell<sup>10</sup> a regroupé dans un livre passionnant, des textes originaux concernant les origines des ordinateurs, et la plupart des informations de ce paragraphe sont extraites de ce livre.

Depuis la *Calculating Clock* de Schickard et la Pascaline de Pascal au début du XVII<sup>ème</sup> siècle, en passant par la machine de Leibniz, et l'*Analytical Engine* de Charles Babbage au début du XIX<sup>ème</sup> siècle, de nombreuses machines à calculer, plus ou moins dédiées à des calculs particuliers et plus ou moins programmables, ont été construites.

C'est sans doute parmi les concepteurs des nombreuses machines à calculer qui, pour des raisons militaires, furent construites pendant la deuxième guerre mondiale qu'on trouvera l'inventeur ou l'innovateur<sup>11</sup> du « premier ordinateur », innovation radicale du XX<sup>ème</sup> siècle. Les trois machines les plus fréquemment citées ont été construites dans trois pays ayant joué un rôle majeur pendant cette guerre, les États-Unis, la Grande Bretagne et l'Allemagne<sup>12</sup>. En Allemagne, Konrad Zuse et les machines Z3 et Z4, aux États-Unis, l'ENIAC de Mauchly et Eckert à la *Moore School of Electrical Engineering* de l'Université de Pennsylvanie, en Grande Bretagne, le Colossus construit par Newman à Bletchey Park.

L'ENIAC, similaire au Colossus mais plus rapide, plus flexible et, surtout capable d'exécuter n'importe quel programme enregistré dans sa mémoire (on dit alors que la machine est Turing-complète), ce que le Colossus n'était pas, pas plus que les machines de Zuse, est souvent considéré comme *the first electronic programmable computer built*.

Mais alors pourquoi considère-t-on généralement que Turing et von Neumann sont les inventeurs de l'ordinateur ? Certes, on peut considérer que ce qu'on appelle, encore aujourd'hui, « architecture de von Neumann », et qui est le schéma général d'un ordinateur, n'est qu'une réalisation de la machine universelle de Turing.

La figure 3 est un schéma de l'architecture de von Neumann, qu'il serait plus correct d'appeler architecture de Turing-von Neumann.

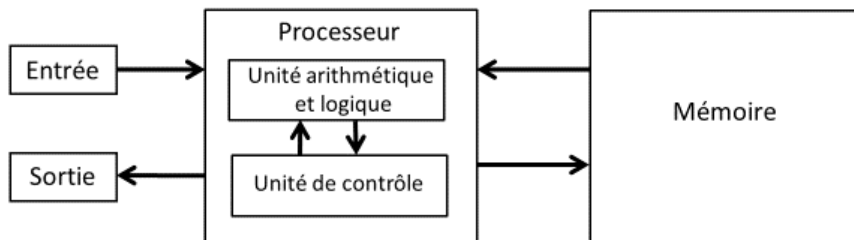


Figure 3. Architecture de Turing-von Neumann

<sup>10</sup> B. RANDELL, édit., *The Origins of Digital Computers*, Springer-Verlag, 1973

<sup>11</sup> Moins usuel en français qu'en anglais au point que dans une déclaration d'Obama concernant Steve Jobs *innovator* est généralement traduit par inventeur (cf. <https://obamawhitehouse.archives.gov/blog/2011/10/05/president-obama-passing-steve-jobs-he-changed-way-each-us-sees-world>)

<sup>12</sup> On vient de « découvrir » le rôle important de cryptanalystes polonais dans le décodage d'Enigma, et il est possible que l'on découvre que des « ordinateurs » avaient été construits en URSS pendant cette période.

Le seul apport de Turing serait-il sa machine universelle ? Et la participation de von Neumann au projet de l'ENIAC aurait-elle été sa seule contribution ?

## L'autre machine de Turing

(Quelque part en Grande-Bretagne en 1944)

– Hello, sir ! dit Alan Turing en retrouvant John von Neumann, qui désirait le voir, dans un endroit que nous garderons secret.

– Hi, Alan ! Comment allez-vous ? Que de choses se sont passées depuis Princeton, n'est-ce pas ? C'est vrai que je viens de fêter mes 40 ans mais appelez-moi quand même John, ne me vieillissez pas trop vite s'il vous plaît, la guerre y pourvoit suffisamment rapidement. Que faites-vous, en ce moment ?

– Secret ! Et vous ?

– Top secret, bien sûr. Enfin, pas tout. Vous avez entendu parler de l'ENIAC de Mauchly et Eckert à la Moore School ?

– Bien sûr.

– Et bien, j'ai rejoint leur projet et je suis en train de préparer la suite, l'EDVAC une machine à calculer à programme enregistré faite avec des tubes à vide, une réalisation de votre machine de 36.

– Vous avez lu mon article de 1936 sur le problème de décision ?

– Naturellement Alan, auriez-vous oublié ? J'avais assisté à votre séminaire chez Alonzo. J'avais trouvé votre solution du *Entscheidungsproblem* très élégante, votre machine universelle aussi, mais un peu abstraite, non ?

– Vous avez raison. Cependant, l'abstraction ...

– est bien mon domaine de prédilection, bien sûr, mais nous avons discuté quelques instants, tous les trois avec Alonzo, des possibilités de construction d'une telle machine, vous ne vous en souvenez-vous ?

– Non. Je n'avais pas dû beaucoup parler. J'étais timide à l'époque et vous m'impressionniez.

– Et maintenant ?

– ...

– Je plaisante Alan, voyons, je plaisante, c'est une des rares choses qu'il nous reste en ce moment, non ? Bon, je vous raconte rapidement ce que je prévois pour l'EDVAC puis vous me dites un mot de vos travaux actuels, OK ?

– Je vous écoute.

(John von Neumann lui décrit les grandes lignes de ce qui sera rédigé dans son célèbre rapport de 1945<sup>13</sup>)

– A votre tour maintenant, Alan.

– Comme vous m'avez parlé de l'EDVAC, je peux vous dire un mot de l'ACE (*Automatic Computing Machine*) dont je suis en train de préciser les spécifications<sup>14</sup>.

(Alan Turing lui expose avec beaucoup de détails ce qui fera l'objet de son rapport peu connu de 1945 et lui apprend qu'il rejoindra sans doute très bientôt le projet Manchester au National Physical Laboratory pour participer à la construction d'une ACE.)

<sup>13</sup> J. von NEUMANN, *First Draft of a Report on the EDVAC*, Contract N° W-670-OKD-4926 Between the United States Army Ordnance Department and the University of Pennsylvania. (Reproduit dans le livre cité de Randell.)

<sup>14</sup> Rapport dont la référence est donnée dans la note 1

- Et bien, je vois que nous travaillons dans à peu près les mêmes directions en approfondissant votre modèle de machine de 1936, c'est bon signe !
- Si Alonzo était avec nous il nous dirait sûrement : n'oubliez pas le métier à tisser de Jacquard et l'*Analytical Machine* de Babbage ! dit Alan en souriant (pour la première fois).
- C'est amusant qu'un logicien comme lui semble fasciné par les machines à calculer !
- Et vous ?
- Tout m'intéresse, vous le savez bien ! Cependant, mon intérêt pour les machines à calculer efficaces est effectivement plutôt nouveau. C'est l'une des raisons pour lesquelles je suis ici. J'ai été très heureux de pouvoir passer quelques instants avec vous Alan. J'espère que la guerre sera bientôt finie et que vous pourrez venir à Princeton discuter de sujets passionnants comme la biologie ou ... les échecs, conclut von Neumann dans un grand éclat de rire.
- Bon retour aux États-Unis !
- Portez-vous bien Alan !

Les deux rapports, celui sur l'EVAC et celui sur l'ACE, ont beaucoup de points communs. Une comparaison technique très détaillée est faite dans l'article « *The other Turing machine* » de Carpenter et Doran<sup>15</sup>.

Pour Carpenter et Doran : « les rapports de 1945 de von Neumann et de Turing sont les premières tentatives de description de l'architecture d'un ordinateur, cependant à des niveaux différents. Le rapport de von Neumann est un brouillon incomplet ... ». Ces deux auteurs notent par exemple que dans la proposition concernant l'EDVAC, il n'y a aucun mécanisme d'entrée/sortie et que la description de l'organe de contrôle est très succincte. « D'un autre côté, écrivent-ils, le rapport de Turing est la description complète d'un ordinateur, jusqu'aux diagrammes des circuits logiques, une analyse exhaustive de 13 pages des propriétés physiques de la mémoire et une estimation du coût : 11. 200 livres ... ». Leur conclusion est la suivante : « En fait, la proposition de Turing est très probablement (*quite possibly*) la première conception de l'architecture d'un ordinateur (*stored program computer architecture*). »

De plus, le rapport de Turing contient également de nombreuses idées qui seront mises en œuvre dans des ordinateurs commerciaux et auraient probablement pu faire l'objet de plusieurs brevets<sup>16</sup>.

Par sa machine universelle, son rôle dans Colossus (l'article de Turing de 1936 a eu une influence explicite auprès de Newmann, responsable du Colossus) et sa proposition concernant l'ACE Turing serait donc notre « vainqueur » ? De son côté, von Neumann, après sa proposition concernant l'EDVAC, eut un rôle important dans la conception de l'EDSAC de Mauchly et Eckert, puis de l'IAS prototype de l'IBM 701<sup>17</sup>, serait un sérieux challenger. Chercher un vainqueur est un peu futile, de très nombreuses personnes (ayant eu de nombreuses collaborations sur différents projets, et de nombreuses rencontres et discussions) ayant participé à cette aventure collective, des chercheurs et des ingénieurs compétents dans de nombreux domaines.

Deux étant exponentiellement supérieur à un – trois aussi, mais « Turing-Mauchly&Eckert-von Neumann », même si plus respectueux des contributions de

<sup>15</sup> B. E. CARPENTER, R. W. DORAN, *The other Turing machine*, The Computer Journal, Volume 20, Issue 3, pages 269–279

<sup>16</sup> Cf la table 2 *Formative ideas in Turing's report* de l'article de Carpenter et Doran cité précédemment.

<sup>17</sup> Rien d'étonnant à ce qu'IBM impose rétrospectivement de décerner le titre de premier ordinateur à celui qui ressemblait le plus à la première machine qu'IBM avait construite !



chacun, ne serait pas très sexy d'un point de vue marketing ... –, on pourrait, au moins, remplacer l'appellation « architecture de von Neumann » par « architecture de Turing-von Neumann ».

Finalement, tout ceci n'est que le résultat d'une longue histoire combinant les calculs (algorithmes), les machines à calculer (ordinateurs), et la manière de les utiliser (programmes), en bref, l'histoire de l'informatique (et le prix le plus important en informatique est le prix Turing). Histoire qui, naturellement, est toujours en chantier (voir le rôle important, récemment popularisé, de chercheurs polonais pour le décryptage d'ENIGMA, à Bletchey Park, par la conception, entre autres, des calculateurs *Bomba* précurseurs de Colossus).

En cherchant à résoudre le problème de décision posé par Hilbert, Turing inventa l'ordinateur. On pourrait donc, pour reprendre une notion à la mode parler de sérendipité pour cette invention. En effet, était présent dans cet article l'idée fondamentale qui distingue un ordinateur d'une machine à calculer et qui est celle de programme enregistré. La mémoire contient programme et données ce qui est très clair dans la machine universelle de Turing. Pourrait-on faire un parallèle avec la découverte de la pénicilline, souvent présentée comme un exemple prototypique de sérendipité ? Mis à part leur importance, ces deux découvertes ont au moins un point commun ; dans les deux cas un seul nom reste associé : Fleming<sup>18</sup> dans un cas, von Neumann dans l'autre, alors que d'autres personnes ont participé à cette découverte et que le premier inventeur est peut-être inconnu ou oublié<sup>19</sup>.

---

<sup>18</sup> Pour la découverte de la pénicilline, trois personnes ont obtenu le prix Nobel de médecine en 1945 : Alexander Fleming, Howard Walter Fowley et Ernst Boris Chain (un cousin de mon père, Chain et Chein n'étant que deux écritures dans l'alphabet latin du nom hébreu composé des deux lettres Het, Noun. Ceci n'est qu'une anecdote, qui n'est peut-être pas sans rapport avec certains propos de cet article, mais mon admiration pour ces deux géants de la pensée que sont Alan Turing et John von Neumann est réellement immense !)

<sup>19</sup> J. POUILLARD, *Une découverte oubliée : la thèse de médecine du docteur Ernest Duchesne (1874-1912)*, HISTOIRE DES SCIENCES MÉDICALES - TOME XXXVI - No1 – 2002, 11-20